

IN THE CLAIMS**RECEIVED
CENTRAL FAX CENTER****NOV 28 2006**

Please amend the claims as follows.

For the Examiner's convenience, a list of all claims is included below.

Claims:

1. (Previously Presented) A method for execution by a microprocessor in response to receiving a single instruction, the method comprising:

receiving the single instruction;

receiving a first vector having a first plurality of numbers and a second vector having a second plurality of numbers, each of the first plurality of numbers pointing to one of a plurality of entries, each of the plurality of entries being in one of a plurality of look-up tables; and

replacing simultaneously the plurality of entries in the plurality of look-up tables with the second plurality of numbers;

wherein the receiving and the replacing operations are performed in response to the microprocessor receiving the single instruction;

wherein the microprocessor comprises a media processor integrated with a memory controller for host memory on a single integrated circuit.
2. (Previously Presented) A method as in claim 1 wherein the first vector having the first plurality of numbers is received from a first entry in a register file; and the second vector having the second plurality of numbers is received from a second entry in the register file.
3. (Original) A method as in claim 2 wherein the single instruction specifies indices of the first and second entries in the register file.

4. (Currently Amended) A method for execution by a microprocessor in response to receiving a single instruction, the method comprising:
receiving the single instruction having an identity number code that specifies a DMA controller and a bit segment which specifies a count indicating a number of entries to be loaded in each of a plurality of look-up units; and
replacing at least one entry in at least one of [[a]] the plurality of look-up units in a microprocessor unit with at least one number using [[a]] the Direct Memory Access (DMA) controller;
wherein the replacing is performed in response to the microprocessor receiving the single instruction.
5. (Currently Amended) A method for execution by a microprocessor in response to receiving a single instruction, the method comprising:
receiving the single instruction having an identity number code that specifies a DMA controller and a bit segment which specifies a count indicating a number of entries to be loaded in each of a plurality of look-up units; and
replacing at least one entry for each of a plurality of look-up units in a microprocessor with a plurality of numbers using [[a]] the Direct Memory Access (DMA) controller;
wherein the replacing is performed in response to the microprocessor receiving the single instruction.
6. (Original) A method as in claim 5 wherein a single index encoded in the instruction specifies a location of the at least one entry in the plurality of look-up units.

7. (Original) A method as in claim 5 wherein a single index encoded in the instruction specifies a total number of the at least one entry for each of a plurality of look-up units.
8. (Previously Presented) A method as in claim 5 wherein a source address of the plurality of numbers in host memory is specified in an entry of a register file.
9. (Original) A method as in claim 8 wherein the single instruction specifies an index of the entry in the register file.
10. (Canceled)
11. (Previously Presented) A method for execution by a microprocessor in response to receiving a single instruction, the method comprising:
receiving the single instruction;
receiving a first vector having a plurality of numbers;
partitioning look-up memory into a plurality of look-up tables;
looking up simultaneously a plurality of elements of a second vector from the plurality of look-up tables, each of the plurality of elements being in one of the plurality of look-up tables and being pointed to by one of the plurality of numbers;
wherein the partitioning and the looking-up operations are performed in response to the microprocessor receiving the single instruction.
12. (Previously Presented) A method as in claim 11 wherein the receiving the first vector having a plurality of numbers comprises:

partitioning a string of bits into a plurality of segments to generate the plurality of numbers.

13. (Original) A method as in claim 12 wherein the single instruction specifies format information in which the plurality of numbers are stored in the string of bits.
14. (Original) A method as in claim 11 wherein the look-up memory comprises a plurality of look-up units, and wherein said partitioning look-up memory comprises:
configuring the plurality of look-up units into the plurality of look-up tables.
15. (Previously Presented) A method as in claim 12 wherein the string of bits is received from an entry of a register file.
16. (Original) A method as in claim 15 wherein the single instruction specifies an index of the entry.
17. (Previously Presented) A method as in claim 11 further comprising:
storing the second vector having the plurality of elements in an entry of a register file.
18. (Original) A method as in claim 17 wherein the single instruction specifies an index of the entry.
19. (Original) A method as in claim 17 wherein the single instruction specifies format information in which the plurality of elements are stored in the entry.

20. (Currently Amended) A method as in claim 11, wherein the look-up memory comprises a plurality of look-up units, and wherein said partitioning look-up memory comprises:
configuring the plurality of look-up units into the plurality of look-up tables;
wherein each of the plurality of look-up units comprises 256 8-bit entries.
21. (Original) A method as in claim 11 wherein the single instruction specifies a total number of entries contained in each of the plurality of look-up tables.
22. (Original) A method as in claim 21 wherein the total number of entries is one of:
a) 256;
b) 512; and
c) 1024.
23. (Original) A method as in claim 11 wherein the single instruction specifies a total number of bits used by each entry contained in the plurality of look-up tables.
24. (Original) A method as in claim 21 wherein the total number of bits is one of:
a) 8;
b) 16; and
c) 24.
25. (Previously Presented) A machine readable media containing an executable computer program instruction which when executed by a digital processing system causes said system to perform a method comprising:
receiving a single instruction;

receiving a first vector having a first plurality of numbers and a second vector having a second plurality of numbers, each of the first plurality of numbers pointing to one of a plurality of entries, each of the plurality of entries being in one of a plurality of look-up tables; and
replacing simultaneously the plurality of entries in the plurality of look-up tables with the second plurality of numbers;
wherein the receiving and the replacing operations are performed in response to the microprocessor receiving the single instruction;
wherein the microprocessor comprises a media processor integrated with a memory controller for host memory on a single integrated circuit.

26. (Original) A media as in claim 25 wherein the first plurality of numbers are received from a first entry in a register file; and the second plurality of numbers are received from a second entry in the register file.
27. (Original) A media as in claim 26 wherein the single instruction specifies indices of the first and second entries in the register file.
28. (Currently Amended) A machine readable media containing an executable computer program instruction which when executed by a digital processing system causes said system to perform a method comprising:
receiving the single instruction having an identity number code that specifies a DMA controller and a bit segment which specifies a count indicating a number of entries to be loaded in each of a plurality of look-up units; and

replacing at least one entry in at least one of a plurality of look-up units in a microprocessor unit with at least one number using a Direct Memory Access (DMA) controller;
wherein the [[the]] replacing is performed in response to the microprocessor receiving the single instruction.

29. (Currently Amended) A machine readable media containing an executable computer program instruction which when executed by a digital processing system causes said system to perform a method comprising:
receiving the single instruction having an identity number code that specifies a DMA controller and a bit segment which specifies a count indicating a number of entries to be loaded in each of a plurality of look-up units; and
replacing at least one entry for each of a plurality of look-up units in a microprocessor with a plurality of numbers using a Direct Memory Access (DMA) controller;
wherein the receiving is performed in response to the microprocessor receiving the single instruction.
30. (Original) A media as in claim 29 wherein a single index encoded in the instruction specifies a location of the at least one entry in the plurality of look-up units.
31. (Original) A media as in claim 29 wherein a single index encoded in the instruction specifies a total number of the at least one entry for each of a plurality of look-up units.
32. (Original) A media as in claim 29 wherein a source address of the plurality of numbers is specified in an entry of a register file.

33. (Original) A media as in claim 32 wherein the single instruction specifies an index of the entry in the register file.
34. (Canceled)
35. (Previously Presented) A machine readable media containing an executable computer program instruction which when executed by a digital processing system causes said system to perform a method comprising:
- receiving a single instruction;
 - receiving a first vector having a plurality of numbers;
 - partitioning look-up memory into a plurality of look-up tables;
 - looking up simultaneously a plurality of elements of a second vector from the plurality of look-up tables, each of the plurality of elements being in one of the plurality of look-up tables and being pointed to by one of the plurality of numbers;
- wherein the partitioning and the looking-up operations are performed in response to the microprocessor receiving the single instruction.
36. (Previously Presented) A media as in claim 35 wherein said receiving the first vector having a plurality of numbers comprises:
- partitioning a string of bits into a plurality of segments to generate the plurality of numbers.
37. (Original) A media as in claim 36 wherein the single instruction specifies format information in which the plurality of numbers are stored in the string of bits.

38. (Original) A media as in claim 35 wherein the look-up memory comprises a plurality of look-up units, and wherein said partitioning look-up memory comprises:
configuring the plurality of look-up units into the plurality of look-up tables.
39. (Previously Presented) A media as in claim 36 wherein the string of bits is received from an entry of a register file.
40. (Original) A media as in claim 39 wherein the single instruction specifies an index of the entry.
41. (Previously Presented) A media as in claim 35 wherein the method further comprises:
storing the second vector having the plurality of elements in an entry of a register file.
42. (Original) A media as in claim 41 wherein the single instruction specifies an index of the entry.
43. (Original) A media as in claim 41 wherein the single instruction specifies format information in which the plurality of elements are stored in the entry.
44. (Original) A media as in claim 38 wherein each of the plurality of look-up units comprises 256 8-bit entries.
45. (Original) A media as in claim 35 wherein the single instruction specifies a total number of entries contained in each of the plurality of look-up tables.

46. (Original) A media as in claim 45 wherein the total number of entries is one of:
- a) 256;
 - b) 512; and
 - c) 1024.
47. (Original) A media as in claim 35 wherein the single instruction specifies a total number of bits used by each entry contained in the plurality of look-up tables.
48. (Original) A media as in claim 47 wherein the total number of bits is one of:
- a) 8;
 - b) 16; and
 - c) 24.
49. (Previously Presented) A method as in claim 5 wherein the at least one entry for each of the plurality of look-up units comprises a plurality of entries for each of the plurality of look-up units.
50. (Previously Presented) A method as in claim 11 wherein the microprocessor comprises a media processor integrated with a memory controller for host memory on a single integrated circuit.